UNIX

32 versus 64 bit Computers

On all computers TART2002 uses 64 bit precision for REAL variables. 64 bit precision means DOUBLE PRECISION on 32 bit UNIX workstations, and SINGLE PRECISION on 64 bit CRAY computers. The only difference between the UNIX and CRAY/64 Bit versions of TART2002 is that the UNIX version uses DOUBLE PRECISION functions, e.g., DEXP, DCOS, DABS, etc., and the CRAY version uses SINGLE PRECISION functions, e.g., EXP, COS, ABS, etc.

WARNING - because of this difference the UNIX and CRAY versions are not interchangeable between computers. On UNIX computers, ONLY use the UNIX version, and on CRAY/64 Bit computers, ONLY use the CRAY version.

Installation of System Sources

1) Make sure you do not have a directory named TART2002, otherwise installation will overwrite it.
2) Copy **TARTZ.Z** and **INSTALL.BAT** from the TART2002 CD **UNIX** directory to your UNIX computer in your base directory [WARNING - do not put them in any other directory]. Make sure the names are EXACTLY as shown here [all upper case characters]
3) Make sure **INSTALL.BAT** is executable,i.e., type, chmod 777 INSTALL.BAT
4) Execute **INSTALL.BAT**.
5) Once installation is finished you can delete **INSTALL.BAT** and **TARTZ.Z**.
6) Copy the **DOCUMENT** directory from the TART2002 CD to the **TART2002** directory on your computer.

Installed Directory Structure

You should now have a main directory named TART2002, and within this directory you should have the following sub-directories,

Documentation

DOCUMENT - On-line documentation in Microsoft Word format

Installation Files

INSTALL - Installation files for a variety of UNIX computers

Production Codes

TART2002 - TART2002 source code
UTILITY - A variety of useful utility codes
TARTDATA - Codes and data to create TART2002 data files

EXAMPLES	- Example TART2002 input parameters
CRITS	- Example critical assembly TART2002 input parameters

Graphics Codes

TARTCHEK	- TARTCHEK source code
TARTAID	- TARTAID source code
IMAGES	- IMAGES source code
EPICDATA	- Codes and data to create EPICSHOW data files
EPICSHOW	- EPICSHOW source code
PLOTTAB	- PLOTTAB source code
EDITOR	- FORTRAN code editor

Implementation

Here's a brief outline of what you have to do to implement the system,

Production Codes

1) Create the binary, random access files used by TART2002
2) Compile TART2002
3) Compile the utility codes

Interactive Graphics Codes

4) Create the binary, random access files used by EPICSHOW
5) Compile and load TARTCHEK, EPICSHOW, IMAGES, PLOTTAB, and EDITOR

INSTALL is designed to do all of these steps for you automatically. As a backup the appendix includes details of how to manually implement the system. The directory TART2002/INSTALL contains install files for a variety of computers. Select the file that applies to your computer, e.g., on SUN use INSTALL.SUN, on HP use INSTALL.HP, etc.

To automatically implement the system,

1) Copy your **INSTALL** file from your **TART2002/INSTALL** directory to your base directory, e.g., for SUN,
   cp TART2002/INSTALL/INSTALL.SUN .
2) Make sure it is executable, ,i.e., type, chmod 777 INSTALL.SUN
3) Execute **INSTALL.SUN**

It can take some time to complete all of these steps (hours), so be patient until it finishes.

Compilation Errors and Warning Messages

While compiling ALL of the codes you should not receive ANY compiler Error messages. While compiling TART2002 you should receive a few Warnings that certain parts of the code cannot be reached - this is o.k. - these are parts of the code that are currently inactive. On some computers you will get MANY Warning messages that some variables are not being used - this is o.k. - your compiler is merely listing all of the variables from the INCLUDE files that may not be used by individual routines.

If you receive any other Error or Warning messages please report them to me in detail.

Optimization Level

Depending on how your computer is configured you may receive compiler/loader messages that compilation or loading is aborted due to exceeding your systems resources; this message is stated in different ways on various computers, but the intent on all computers is the same. There are several ways to correct this problem: a) The simplest way is to decrease the optimization level in your Makefile, e.g., change -O5 to -O2. This is not recommended, since reducing optimization will increase execution time, b) Have your system supervisor increase the system resources allowed for compilation/loading. This is the recommended procedure, but may not be practical if nobody at your installation knows how to do this.

Verification

It is highly recommended that you not be in too much of a rush to start using the system, and instead spend the time to FIRST verify that TART2002 is producing reliable results.

The example problem is in the TART2002/TART2002 directory ready to be used with TART2002. It is HIGHLY RECOMMENDED that you take the time to first run this problem. To start the problem type,

```
tart02-4
```

The execution time will be somewhere between 100 and 18,000 seconds, (about 2 minutes to 5 hours), depending on the speed of your computer. When the problem ends, copy the utility code, critedit, from the utility directory to the current directory and execute critedit by typing,

```
critedit
```

This will give you a summary of the results of the problems run. The important things to check are at the bottom of the output listing, on your screen, and in the file CRITEDIT.LST.

First check the running time, which will give you a good indication of the relative power of your computer - see TART2002.DOC for a list of expected times on a wide variety of computers.

Next check the calculated Average K-eff - this should be about 0.999… to 1.001…

Cleaning Up

Once you have verified that everything is installed and working properly you can delete all of the system source files,

1) Copy **CLEANUP.BAT** from your **TART2002/INSTALL** directory to your base directory,

cp TART2002/CLEANUP.BAT .

2) Make sure it is executable, ,i.e., type, chmod 777 CLEANUP.BAT
3) Execute **CLEANUP.BAT**

APPENDIX

The following is provided as a backup, in case **INSTALL** file has any problems in compiling and installing the codes. If here is a problem you can still follow the below steps.

UNIX vs. Other System Users

The following instructions are for UNIX users; this includes all variants of UNIX, such as LINUX, etc. For users of other systems first see the instructions in the OTHER section of this installation manual, for information on computer dependence. You can then return to this point and follow these steps - modified to meet your computer's needs.

Production Codes

1) In TARTDATA make sure the file MAKEIT.BAT is executable, i.e., type,

```
chmod 777 MAKEIT.BAT
```

Then execute it by typing,

```
MAKEIT.BAT
```

This will compile four codes and then run each code to create the four binary, random access files used by TART2002.

On most computers there is nothing to do until this batch file finishes running. However, on some computers you will be asked to approve deletion of certain files - answer yes (y) each time.

When this batch file has finished running you will find the four random access, binary files,

TARTND	- Neutron interaction data
TARTPPD	- Neutron induced photon production data

GAMDAT	- Photon interaction data
NEWCROSS	- Neutron multi-band, self-shielding data

Move these four files to the directory where you will execute ~/TART2002/TART2002.

Once you have created the binary, random access data files and moved them to where you will be using them you no longer need the TARTDATA directory and you can delete it.

2) In TART2002, you next have to compile TART2002 in the directory TART2002/TART2002/SOURCES. There are still a few computer dependent parts of TART2002, e.g., execution time, date, etc. All of these computer independent parts are isolated in files named ???.f - where ??? is the name of a specific type of computer, e.g., DEC.f, SUN.f, HP.f, etc.

Makefiles are provided for each type of computer. The only differences between the various Makefiles are: a) they will use the correct ???.f file for each type of computer, b) they will compile on each computer using the highest level of optimization defined by each computer. The Makefiles use a similar name convention, e.g., SUN.mak, HP.mak, etc.

To compile all you need do is select the Makefile for your computer, copy it to Makefile, e.g., for use on a SGI,

```
cp SGI.mak Makefile
```

Make sure Makefile is executable and then make the executable by typing,

```
make
```

Next copy tart02-4 to ~/TART2002/TART2002 sub-directories where you will be executing the code.

Once you have done this you no longer need the source code, Makefiles, etc. in this directory and you can delete the directory TART2002/TART2002/SOURCES.

When you have completed this step you will be able to run tart02-4. To run the example problem type,

```
 tart02-4
```

3) In UTILITY, make sure the file INSTALL.BAT is executable, e.g.,

```
chmod 777 INSTALL.BAT
```

Then execute it by typing,

INSTALL.BAT

This will compile all of the utility codes for your later use. See the documentation of the utility codes for details on how to use the utility codes.

Once you have done this you no longer need the source code, in this directory and you can delete all *.f and INSTALL.BAT.

Interactive Graphics Codes

32 versus 64 bit

The interactive graphics codes are **currently** only designed to be used on 32 bit computers. If you are using a CRAY or other 64 bit computer you will not be able to use these codes. However, these codes offer such advantages that even if your primary computer is a CRAY or 64 bit computer, if you have access to any UNIX workstations, IBM-PC, or POWERMAC I highly recommend that you implement these codes. These codes can save you an enormous amount of time and energy in preparing and testing TART2002 input, and in analyzing TART2002 output.

32 bit

In principle, for these codes you will be able to use simple procedures similar to those used above to install TART2002. In practice interactive graphics are still not completely standard and you may have more work to do. Since any problems that you may encounter apply to all of the following codes let go over them once here - once you solve these problems for one of the codes, all the others will be simple plug-ins. Here's a list of the problems you may run into and how to solve them,

1) The codes use the standard UNIX X11 graphics library. This is a standard library, but it isn't always stored in the same place on all computers - so it may not be where the enclosed Makefiles are looking for it. Therefore you may have to modify the Makefiles to direct them to where X11 is stored on your system - or alternatively, copy the X11 library to where the
Makefiles are looking for them.

2) The graphics interface is written in C and the FORTRAN to C interface is not the same on all computers. On some computers all C routine names end in underscore (_), while on others they do not. Routines are supplied to handle both conventions,

dash.c	= includes _ at the end of names
nodash.c	= does not include _

If during loading you receive error messages that many routines are missing, look in the Makefile you are using to see whether you are using either, dash.c or nodash.c - and try using the other one.

3) Lastly in the case of TARTCHEK you should define running time. If you have gotten this far you have already gotten TART2002 running, including a timing routine - so you can use the same procedure here. The only difference is that TART2002 defines time since the beginning of execution, whereas TARTCHEK defines time between calls to the timing routine - for details, see the timer routines in the TARTCHEK directory.

Solving these problems may not be that simple - particularly the problem of finding the X11 graphics system on your computer and modifying the Makefiles to access X11. However, be assured that this system has been installed on many, many UNIX systems and we have NEVER found one that we couldn't implement these codes on. Some systems are more difficult than others, but eventually we have gotten the codes to work ON ALL OF THEM! So hang in there and keep trying - it can be done, and having these codes available is worth the effort NOW, because they will make your life a lot easier in the FUTURE.

How do you find out if you have any of these problems? Try compiling TARTCHEK and you'll find out rather quickly.

Below I assume you have solved all of these problems and I will discuss implementation of the remaining codes.

4) In EPICDATA, make sure the file INSTALL.BAT is executable, e.g.,

```
chmod 777 MAKEIT.BAT
```

Then execute it by typing,

```
INSTALL.BAT
```

This will compile a code and then run it to create the four, binary, random access files used by EPICDATA.

On most computers there is nothing to do until this batch file finishes running. However, on some computers you will be asked to approve deletion of certain files - answer yes (y) each time.

When this batch file has finished running you will find the four random access, binary files,

NEUTRON.BIN	- Neutron interaction data
PHOTON.BIN	- Photon interaction data
ELECTRON.BIN	- Electron interaction data
CHARGED.BIN	- Light Charged Particle interaction data

Move these four files to the directory where you will execute EPICSHOW, e.g., ~/TART2002/EPICSHOW.

Once you have created the binary, random access data files and moved them to where you will be using them you no longer need the EPICDATA directory and you can delete it.

5) The procedure for implementing the graphics codes is the same for all six graphics codes,

TARTCHEK
TARTAID
IMAGES
EPICSHOW
PLOTTAB
EDITOR

There is a directory for each, and within each of these directories there is a sub-directory named SOURCES. As with TART2002, these directories contain the basic source code, plus computer dependent code for each type of computer, and Makefiles for each type of computer. In each case all you have to do is select the Makefile for your type of computer, copy it to Makefile, e.g., for use on an SGI,

```
cp SGI.mak Makefile
```

and then compile and load by typing,

```
make
```

Once you have compiled and loaded a code, move it from the SOURCES directory, e.g., for tartchek, move it from TARTCHEK/SOURCES to TARTCHEK. In the case of PLOTTAB remember to move both plottab (on screen output) and plotsave (Postscript, hardcopy output). When you have done this you no longer need the source code, Makefiles, etc. and you can delete the SOURCES directory.